

عن المؤلف:

الإسم:أحمد ولد محمد

اللقب: المتألق (elmoute2eli9)

سنة الميلاد:1992

الدولة:موريتانيا

الهواية: البرمجة و الهاكينج

المستوى:طالب جامعي بكلية العلوم و التقنيات

للاستفسار و تبادل الخبرات: elmoute2eli9@hotmail.com

صورة المؤلف:





أُهدي هذا الكتاب إلى أخي و حبيبي الغالي chikoo و إلى أختي الأكبر zinat و إلى أستاذي المحترف sembawyo كما أُهديه أيضا إلى منتديات العاصفة و منتديات الفريق العربي للبرمجة.

أسباب تأليف الكتاب:

نظرا لكثرة الأسئلة عن استخدام الدالة system و هي دالة في غاية الأهمية فقد قررت أن أقوم بشرح مفصل لها لكي يستوعبها المبتدئون بشكل أفضل.

البداية:

لنبدأ على بركة الله

أولا لنلق نظرة على الكود التالي:

```
/*
  A propos du programmeur:
  Nom:Ahmed/Mohamed
  Prénom:elmoute2eli9 (المتألق)
  Année de naissance:1992
  E-mail:elmoute2eli9@hotmail.com
  Description:L'utilisation avancé de la fonction system
*/
#include<iostream>
using namespace std;
int main()
{
    cout<<"by elmoute2eli9"<<endl;
    system("pause"); السطر الرابع عشر
    return 0;
}
```

طبعا هذا الكود يظهر الجملة : by elmoute2eli9 على الشاشة.

لنرجع قليلا إلى السطر الرابع عشر:

```
system("pause");
```

*لماذا كتبنا العبارة السابقة ؟؟؟

الهدف من العبارة السابقة هو إيقاف البرنامج حتى نَتَمَكن مِن مُشاهدة مخرجات البرنامج .. فكما تعلم أن الحاسوب بإمكانه القيام بآلاف العمليات الحسابية و المنطقية للدقيقة الواحدة بل أكثر!!! و نظرا لبطئ الإنسان مقارنة مع الكومبيوتر فإننا نحتاج إلى إيقاف البرنامج حتى نشاهد النتائج.

*ما هي الدالة system ؟؟؟

الدالة system هي دالة تعمل كوسيط بين المستخدم و نظام التشغيل سواء كان ويندوز أو لينكس أو حتى ماكينتوش !!!

*ما هي فائدة هذه الدالة؟؟؟

لنفرض أنك تستعمل نظام لينكس أو ويندوز و تريد إدخال بعض أوامر الدوس أو أوامر لينكس في برنامجك...عندها ستتولى الدالة system هذه المهمة!!!

كيف؟؟؟ إليك المثال:

نريد الآن مسح شاشة البـــرنامج , طبعا بإمكاننا استخدام الدالة clrscr() ; الموجودة في المكتبة conio.h لكننا سنستخدم الدالة system مع الأمر المناسب و هو cls إذا كُنتَ في الويندوز أو clear إذا كُنتَ في لينكس كالتالي:

```
system("cls");//DOS
system("clear");//LINUX,UNIX
```

*ما هي بارامترات أو وسائط هذه الدالة؟؟؟

الدالة system نُدخل لها وسيط واحد , هذا الوسيط أو البارامتر هو أمـــر من أوامر النظام الذي نود تنفيذه مثلا في المثال الأول (السطر الرابع عشر) قمنا بإدخال الأمر Pause و هو أمر من أوامر الدوس مسئول عن إظهار الجملة :

```
Appuyez sur une touche pour continuer...
```

حيث يظل البرنامج ينتظر... حتى يضغط المستخدم على زر من لوحة المفاتيح حينها يقوم البرنامج بالانتقال إلى الخطوة القادمة وهي:

```
return 0;
```

و تعني الخروج من البرنامج.

الآن أصبح بإمكاننا أن نكتب الكود المكافئ للعبارة التي في السطر الرابع عشر من المثال الأول .. و هذا هو الكود :

```
cout<<"Appuyez sur une touche pour continuer...";
getch();
```



خلاصة:

الدالة system مسئولة عن تنفيذ أوامر النظام حيث تلعب دور الوسيط بين المستخدم و نظام التشغيل.

الآن سنقوم بإعطاء بعض الأمثلة لاستعمال هذه الدالة لكن يجب عليك أولا معرفة (ولو قليلا) لبعض أوامر الدوس مع أني قمت بشرحها !!!

I- المثال الأول:

اكتب برنامج يُظهر كلا من الوقت و التاريخ الحاليين مع إمكانية تغيـــيــــرهما.

في هذا المثال سنستعمل الأمرين time و date, الأمر الأول مسئول عن الوقت أما الثاني فمسئول عن التاريخ و الكود الآتي يوضح ما سبق:

```
#include<iostream>
using namespace std;
int main()
{
    system("time");
    system("date");
    system("Pause");
    return 0;
}
```

أما إذا أردنا عدم تغيير الوقت فنكتب الأمر:

```
time /t
```

بدل الأمر السابق أقصد الأمر time.

II- المثال الثاني:

اكتب برنامج يُظهر أربع خيارات للمستخدم على الشكل التالي:

1-إيقاف تشغيل الجهاز (Arrêter l'ordinateur)

2-إعادة تشغيل الجهاز (Redémarrer l'ordinateur)

3-إنهاء جلسة المستخدم (Fermer la session)

4-الخروج من البرنامج (Quitter le programme)

وعند الضغط على 1 يتم إيقاف الكومبيوتر ,2 تتم إعادة التشغيل و هكذا بالنسبة لبقية الأوامـــر..

في هذا المثال سنحتاج إلى الأوامر التالية :

```
shutdown -s -t 00
shutdown -r -t 00
shutdown -l
exit
```

شرح الأوامر السابقة :

نلاحظ أن هذه الأوامر(طبعا ما عدا الأمر exit) تتكون من الكلمة shutdown و البارامترات -s و -t و -l.

بالنسبة للكلمة shutdown فتعني بالفرنسية : fermeture و بالعربية : إغلاق

أما البارامترات أو الوسائط فهي تابعة لهذا الأمر أقصد الأمر shutdown حيث يتغير دور هذا الأمر بِتغير الوسائط و يحتوي على 10 وسائط !!! , بإمكانك رؤيتها في نافذة الدوس عن طريق كتابة الأمر shutdown ثم الضغط على Enter عندها سترى لائحة تشمل وسائط هذا الأمر مع الشرح...كما في الصورة:

```
Utilisation : shutdown [-i | -l | -s | -r | -a] [-f] [-m \\nom_ordinateur] [-t xx] [-c "commentaire"
] [-d up:xx:yy]

        Sans argument           Affiche ce message (identique à -?)
        -i                      Affiche l'interface graphique, doit être la 1ère option
        -l                      Ferme la session (ne peut pas être utilisé avec l'option -m)
        -s                      Met l'ordinateur hors tension
        -r                      Met l'ordinateur hors tension et le redémarre
        -a                      Annule une mise hors tension système
        -m \\nom_ordinateur     Ordinateur distant à mettre hors tension/redémarrer/annuler
        -t xx                   Définir le délai d'expiration pour la mise hors tension à xx seconde
s
        -c "commentaire"                Commentaire de la mise hors tension (maximum de 127 caractèr
es)
        -f                      Force des applications en cours d'exécution à se fermer sans avertis
sement
        -d [u][p]:xx:yy         Le code de raison de la mise hors tension
                                u est le code utilisateur
                                p est un code de mise hors tension planifié
                                xx est le code de raison majeur (valeur entière inférieure à 256)
                                yy est le code de raison mineur (valeur entière positive inférieure
à 65536)
```

نرجع إلى مثالنا السابق .. وبالضبط إلى البارامترات (الوسائط) :

بالنسبة للأمر الأول:

الوسيط s– : مهمته إغلاق الجهاز.

الوسيط t– : مهمته تحديد الفترة الزمنية قبل تنفيذ الأمـــر حيث تُحسب هذه الفترة بالثواني فإذا أردنا إيقاف الجهاز بعد 3 دقائق نكتب الأمر:

```
shutdown -s -t 180
```

حيث : 3 دقائق = 3*60 ثانية = 180 ثانية .

طيب الآن سيتوقف الجهاز بعد 3 دقائق!!!



كيف أُلغي هذا الأمر؟؟؟

نكتب الأمر:

```
shutdown -a
```

حيث يقوم الوسيط a– بإلغاء تنفيذ الأمر.

بالنسبة للأمر الثاني:

الوسيط r– : مهمته إعادة تشغيل الجهاز , نفس الشيء بالنسبة لإعادة تشغيل الجهاز ... يمكننا أيضا استخدام الوسيط a– لإلغاء تنفيذ الأمـــر.

بالنسبة للأمر الثالث:

الوسيط l– : مهمته إنهاء جلسة المستخدم.

و أخيرا الأمر exit و مهمته الخروج من البـــرنامج.

الآن صار بإمكاننا كتابة الكود:

```
#include<iostream>
using namespace std;
int main()
{
    char choix;
    cout<<endl<<"\t\t*****************************************************"<<endl;
    cout<<"\t\t   ********** Les commands de shutdown *********"<<endl;
    cout<<"\t\t*****************************************************"<<endl<<endl;
    cout<<"1-pour arrêter l'ordinateur."<<endl;
    cout<<"2-pour redémarrer l'ordinateur."<<endl;
    cout<<"3-pour fermer la session."<<endl;
    cout<<"4-pour quitter le programme."<<endl;
    cout<<"donnez votre choix:";
    debut:
    cin>>choix;
    switch(choix)
    {
    case '1': system("shutdown -s -t 00");break;
    case '2': system("shutdown -r -t 00");break;
    case '3': system("shutdown -l");break;
    case '4': system("exit");break;
    default:{cout<<"Erreur !\a choix incorrect ."<<endl;goto debut;}
    }
    return 0;
}
```

الآن سنقوم بإضافة بسيطة للكود السابق... وهي تغيير لون الكتابة و لون الخلفية أيضا !!!

سنستخدم الدالة system و بالضبط الأمر color و هو الأمر المسئول عن تغيير لون الخط و لون الخلفية , الشكل العام للأمر هو : .. color حيث نضع مكان النقاط رقمين من القاعدة الست عشرية (Base hexadécimale) , الرقم الأول يمثِل لَون الشاشة الخلفية أما الثاني فيمثِل لَون الكتِابَة و هذا الجدول يوضح كل رقم مع اللون الذي يقابله :

قمت بكتابة أسماء الألوان باللغة الفرنسية لأنني لا أعرفها جيدا باللغة العربية!!!

الرقم	اللون المقابل	الرقم	اللون المقابل
0	Noir	8	Gris
1	Bleu foncé	9	Bleu clair
2	Vert	A	Vert clair
3	Bleu-gris	B	Cyan
4	Marron	C	Rouge
5	Pourpre	D	Rose
6	Kaki	E	Jaune
7	Gris clair	F	Blanc

إذا أردتَ تفاصيل أكثر عن هذا الأمر فيمكنك الذهاب إلى الدوس و كتابة الأمر help color و ستظهر لك مجموعة من التعليمات تتعلق بشرح هذا الأمر...



```
Change les couleurs par défaut du premier et de l'arrière plan de la console.

COLOR [attr]

  attr    Spécifie les attributs de couleurs de l'apparence de la console

Les attributs de couleurs sont spécifiés par DEUX chiffres hexadécimaux -- le
premier correspond à l'arrière plan, le second au premier plan. Chaque chiffre
peut prendre n'importe quelle de ces valeurs :

    0 = Noir        8 = Gris
    1 = Bleu foncé  9 = Bleu clair
    2 = Vert        A = Vert clair
    3 = Bleu-gris   B = Cyan
    4 = Marron      C = Rouge
    5 = Pourpre     D = Rose
    6 = Kaki        E = Jaune
    7 = Gris clair  F = Blanc

Si aucun argument n'est donné, cette commande restaure les couleurs
sélectionnées au moment où CMD.EXE a été ouvert. Cette valeur vient soit de la
fenêtre de la console, du commutateur en ligne de commande /T, ou de la valeur
DefaultColor du registre.

La commande COLOR met ERRORLEVEL à 1 si vous tentez de l'exécuter
avec la même couleur pour l'arrière et le premier plan.


Exemple : "COLOR fc" affiche du rouge sur du blanc
```


طيب الآن سنقوم بإضافة السطر:

```
system("color 2E");
```

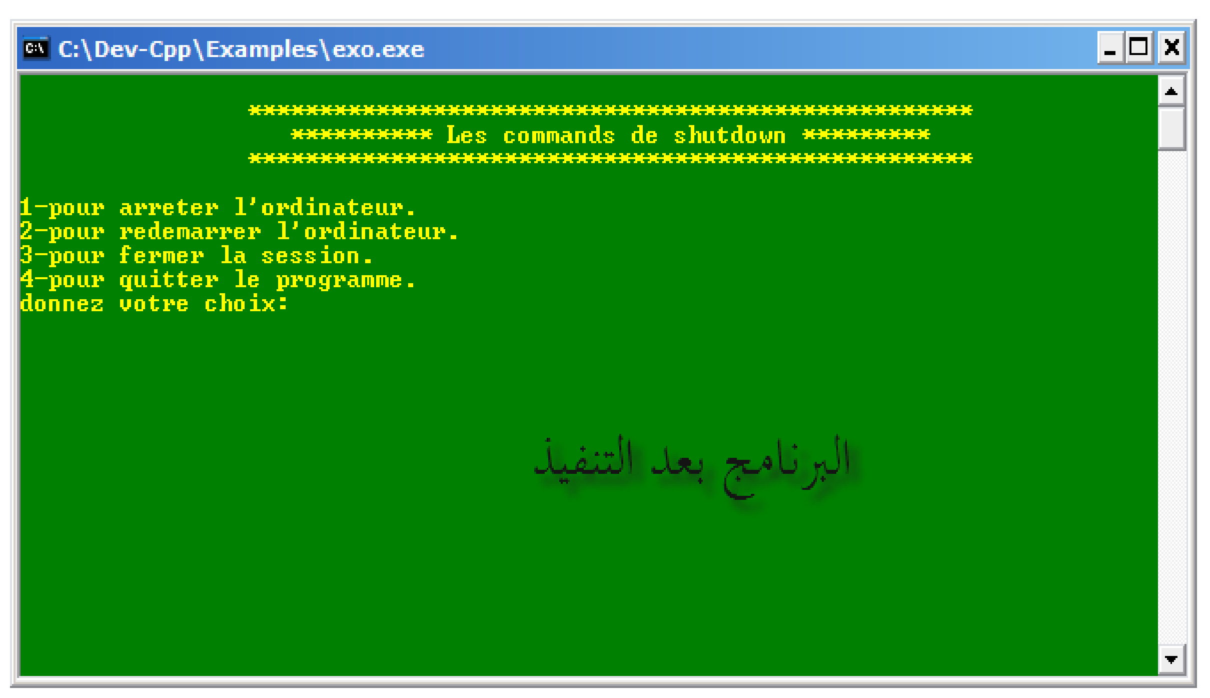
ليصبح الكود هكذا:

```
char choix;
system("color 2E");
cout<<endl<<"\t\t***************************************************"<<endl;
cout<<"\t\t   *********** Les commands de shutdown **********"<<endl;
cout<<"\t\t***************************************************"<<endl<<endl;
```


أنا شخصيا اخترت اللونين الأصفر و الأخضر لأنهما يمثلان عَلَمَ دَولَتِي: موريتانيا

و أنتَ يُمكنُكَ اختيار ما تشاء ..فقط راجع الجدول

و هذه نتائج تنفيذ البرنامج:



أرجو أن تكون الفكرة واضحة ... وإذا وجدت أي استشكال فلا تتردد في مراسلتي!!!

III-المثال الثالث :

اكتب برنامج يقوم بتغيير باسوورد المستخدم!!!

في هذا المثال سنحتاج إلى الأمر net user و هذا الأمر في الحقيقة هو أحد وسائط الأمر net حيث إن الأمر net استخداماته متقدمة و جميلة أيضا فهو يُتيِحُ لَكَ إمكانية إعطاء الحد الأدنى لعدد أحرف كلمة المرور الخاصة بحساب المستخدم و تعيين الحد الأقصى لعدد الأيام التي تكون فيها كلمة المرور الخاصة بحساب المستخدم صالحة و تعيين الحد الأدنى لعدد الأيام قبل أن يتمكن المستخدم من تغيير كلمة المرور و الكثير الكثير...لكنني هنا أردت الاختصار فحسب !!! نرجع للتابع user .. هذا التابع يمتلك 4 وسائط وهو أمر مفيد بإمكانه إضافة مستخدم جديد أو حذفه أو تغيير الباسوورد الخاص به لكننا هنا سنقتصر على تغيير الباسوورد .

الشكل العام للأمر هكذا:

Net user (اسم الحساب) (كلمة المرور)

في هذه الحالة فإن الأمر يستقبل واسطين الأول هو اسم الحساب و الثاني هو كلمة المرور الجديدة:

أظن أنه صار بإمكاننا الآن تغيير الباسوورد.. كما في الكود الآتي:

```
#include<iostream>
using namespace std;
int main()
{
    system("net user %username% 123456");
    system("Pause");
    return 0;
}
```

أرجو أن أكون قد وُفقت في الشرح و أرجو لكم الاستفادة أيضا و لا تنسوني من صالح دعائكم.

انتهى الدرس و إلى لقاء قادم إن شاء الله .. تقبلوا فائق تحياتي.

تم الانتهاء من الكتاب بتاريخ 2009/12/27